Preprints, 10th IFAC International Symposium on
Advanced Control of Chemical Processes
Shenyang, Liaoning, China, July 25-27, 2018

Adaptation strategies for tracking constraints under plant-model mismatch

Martand Singhal [*], **Timm Faulwasser** [**]
and Dominique Bonvin [*]

[*] *Laboratoire d'Automatique, École Polytechnique Fédérale de Lausanne, CH-1015 Lausanne, Switzerland*
{*martand.singhal, dominique.bonvin*}*@epfl.ch*
[**] *Institute for Automation and Applied Informatics, Karlsruhe Institute of Technology, 76344 Eggenstein-Leopoldshafen, Germany,*
timm.faulwasser@kit.edu

Abstract: Optimal operating conditions for a process plant are typically obtained via model-based optimization. However, due to modeling errors, the operating conditions found are often sub-optimal or, worse, they can violate critical process constraints. Hence, model corrections become a necessity and are done by exploiting measured process data. To this end, either model parameters are adapted and/or correction terms are added to the model-based optimization problem. The modifier-adaptation methodology does the latter by adding bias and gradient correction terms that are called *modifiers*. The role of modifiers and model parameters are often seen as competing, and which one of the two is better suited to track the optimality conditions is an open problem. This paper attempts to shed light on finding a synergy between the model parameters and the modifiers in the case when tracking constraints is sufficient for near-optimal performance. We demonstrate through the simulation study of a batch-to-batch optimization problem that a set of model parameters can be selected that mirror the role of modifiers. The modifiers are then added only when there is insufficient number of *mirror parameters* for independent constraint tracking.

Keywords: model-based optimization, real-time optimization, model-plant mismatch, parameter estimation, parameter selection, modifier adaptation

1 INTRODUCTION

Model-based optimization enables process plants to attain optimal performance, while meeting constraints on product quality, environmental regulations and safety. Since models are mere approximations of the actual process, they need corrections that are based on process data. An iterative optimization procedure that attempts to reach plant optimality by model corrections is known as model-based real-time optimization (RTO).

In RTO, there have been several attempts to correct the model via parameter adaptation. The classical two-step approach (Chen and Joseph, 1987) adapts the model parameters to minimize the output error, and new inputs are computed on the basis of the updated model. To find the plant optimum, the whole procedure is repeated until convergence. Recently, Mandur and Budman (2015) proposed to iteratively adapt model parameters in two stages. In the first stage, the output error is minimized, and, in the second stage, the errors on the cost and constraint gradients are minimized.

Due to model complexity, limited information in the process data and the presence of noise, model corrections based on parameter adaptation is often insufficient in capturing the optimal plant behavior (Forbes et al., 1994; Yip and Marlin, 2004; Marchetti, 2009). The plant optimality conditions can be tracked via parameter adaptation if the model offers sufficient flexibility (Chachuat et al., 2009). Since the parameter estimation problem is often nonlinear in parameters and also because of parameter identifiability issues, ensuring sufficient model flexibility is not straightforward. To the best of our knowledge, the current literature does not provide such analysis.

Despite the aforementioned issues, parameter adaptation has some obvious benefits. For instance, adapting model parameters may favorably impact the curvature information, thereby potentially increasing the convergence rate to the optimum (Ahmad et al., 2017). Also, Mandur and Budman (2015) showed through a simulation study that the noise in process data can be better handled by adapting model parameters. Moreover, adapting model parameters is strongly advocated when the model is expected to be structurally correct in the sense that there exists parameter values such that the model and the plant have matching outputs and gradients.

Other real-time optimization approaches such as integrated system optimization and parameter estimation (ISOPE) (Brdyś and Tatjewski, 2005) and modifier adaptation (MA) (Gao and Engell, 2005; Marchetti et al., 2009, 2016) use bias and/or gradient correction terms that can directly be added to the model-based optimization problem. In MA, the added correction terms are

tailored to meet the necessary conditions of optimality, thereby providing complete model flexibility (Chachuat et al., 2009). Hence, MA guarantees meeting the Karush-Kuhn-Tucker (KKT) necessary conditions of optimality for the plant upon convergence. However, the computation of modifiers often requires additional plant experiments to estimate gradients, which makes MA an experimentally expensive methodology. Costello et al. (2016) and Singhal et al. (2017, 2018) proposed to perform a model-based sensitivity analysis to reduce the experimental cost as only a partial correction is made to update the model-based optimization problem. However, since the sensitivity analysis is model based, a model that is far off the reality may not result in sufficient corrections. Moreover, handling noise in MA poses a challenge as it enters the optimization problem via modifiers. Gao et al. (2016) proposed a quadratic-approximation-based MA scheme that requires additional data to mitigate the impact of noise.

Given the pros and cons of adapting model parameters and of adapting modifiers, there is a need for better understanding and, accordingly, establishing synergies between the two. Since the model parameters often do not offer sufficient flexibility, it is desirable to introduce modifiers to compensate for this lack of flexibility. In this paper, we focus only on cases where tracking constraints is sufficient for near optimal performance. We exploit tools such as sensitivity analysis and active subspaces (Constantine et al., 2014; Constantine, 2015) to analyze model flexibility. This way, a set of parameters can be found that behave similar to modifiers. Subsequently, modifiers are added only to constraints that lack so-called *mirror* parameters.

The paper is structured as follows. Section 2 briefly revisits different adaptation strategies and presents the concept of influential and non-influential parameters. Section 3 presents desired characteristics that the adapted parameters should possess and proposes an active-subspace-based procedure that selects parameters that have desired properties. Section 4 summarizes the proposed adaptation strategy. A simulation study is discussed in Section 5. Finally, Section 6 concludes the paper.

2 PRELIMINARIES

2.1 Problem Statement

The optimization of process operation consists in minimizing operating costs, or maximizing economic profit, in the presence of constraints. Mathematically, such problems can be casted as the following optimization problem:

$$\min_{\mathbf{u}} \quad \Phi_p(\mathbf{u}) := \phi(\mathbf{u}, \mathbf{y}_p(\mathbf{u})) \tag{1a}$$

$$\text{s.t.} \quad \mathrm{G}_{p,i}(\mathbf{u}) := \mathrm{g}_i(\mathbf{u}, \mathbf{y}_p(\mathbf{u})) \leq 0, \quad i = 1, \ldots, n_{\mathrm{g}}, \tag{1b}$$

where $\mathbf{u} \in \mathbb{R}^{n_{\mathrm{u}}}$ are the decision (or input) variables; $\mathbf{y}_p \in \mathbb{R}^{n_{\mathrm{y}}}$ are the measured output variables; $\phi : \mathbb{R}^{n_{\mathrm{u}}} \times \mathbb{R}^{n_{\mathrm{y}}} \to \mathbb{R}$ is the cost function to be minimized; $\mathrm{g}_i : \mathbb{R}^{n_{\mathrm{u}}} \times \mathbb{R}^{n_{\mathrm{y}}} \to \mathbb{R}$, $i = 1, \ldots, n_{\mathrm{g}}$, is the set of process-dependent inequality constraint functions. The subscript p refer to a plant quantity. These plant quantities are a priori unknown, but their estimates can be obtained from process data.

Usually, a process model is available, for which the optimization problem reads:

$$\min_{\mathbf{u}} \quad \Phi(\mathbf{u}, \boldsymbol{\theta}) := \phi(\mathbf{u}, \mathbf{y}(\mathbf{u}, \boldsymbol{\theta})) \tag{2a}$$

$$\text{s.t.} \quad \mathrm{G}_i(\mathbf{u}, \boldsymbol{\theta}) := \mathrm{g}_i(\mathbf{u}, \mathbf{y}(\mathbf{u}, \boldsymbol{\theta})) \leq 0, \; i = 1, \ldots, n_{\mathrm{g}}. \tag{2b}$$

where $\boldsymbol{\theta} \in \mathbb{R}^{n_\theta}$ is the vector of model parameters.

2.2 Two-Step Approach

Model parameters can be adjusted so that the plant behavior is well captured. In the two-step approach, plant measurements are used to identify some of the model parameters at the current operating point by solving the following least-squares problem:

$$\boldsymbol{\theta}_k := \arg\min_{\boldsymbol{\theta}} \|\mathbf{y}_p(\mathbf{u}_k) - \mathbf{y}(\mathbf{u}_k, \boldsymbol{\theta})\|^2, \tag{3}$$

where $\|\cdot\|$ is the Euclidean norm; $\mathbf{y}_p(\mathbf{u}_k)$ are the plant measurements at the current operating point $\mathbf{u}_k$, and k represents the iteration number. The updated model parameters are then used in the optimization problem (2) to generate the new set of inputs $\mathbf{u}_{k+1}$. The procedure is repeated until convergence. However, this approach typically does not reach plant optimality in the presence of structural plant-model mismatch (Forbes et al., 1994).

2.3 Constraint Adaptation

Certain optimization problems are mostly driven by the set of active constraints. In these cases, one may reach near optimal performance simply by tracking constraints. For this, an RTO approach known as constraint adaptation (CA) can be used (Chachuat et al., 2008). CA is an iterative scheme that solves the following optimization problem to reach the plant optimum $\mathbf{u}_p^\star$.

$$\min_{\mathbf{u}} \quad \Phi_{m,k}(\mathbf{u}) := \Phi(\mathbf{u}, \boldsymbol{\theta}_0) \tag{4a}$$

$$\text{s.t.} \quad \mathbf{G}_{m,k}(\mathbf{u}) := \mathbf{G}(\mathbf{u}, \boldsymbol{\theta}_0) + \boldsymbol{\varepsilon}_k^{\mathrm{G}} \leq \mathbf{0}, \tag{4b}$$

where $\boldsymbol{\varepsilon}_k^{\mathrm{G}} \in \mathbb{R}^{n_{\mathrm{g}}}$ is the vector of zeroth-order modifier with $\varepsilon_{i,k}^{\mathrm{G}}$ as its i^{th} component; and $\mathbf{G} \in \mathbb{R}^{n_{\mathrm{g}}}$ is the vector of model constraints G_i. At the k^{th} RTO iteration, the modifiers are computed as follows:

$$\boldsymbol{\varepsilon}_k^{\mathrm{G}} = \mathbf{G}_p(\mathbf{u}_k) - \mathbf{G}(\mathbf{u}_k, \boldsymbol{\theta}_0), \tag{5}$$

where $\mathbf{G}_p \in \mathbb{R}^{n_{\mathrm{g}}}$ is the vector of plant constraints $\mathrm{G}_{p,i}$. Note that parameter adaptation is not required in CA as the modifiers introduce bias corrections at each iteration, which suffices to track the plant constraints. Hence, model parameters are fixed at their nominal values $\boldsymbol{\theta}_0$.

CA can yield optimality without requiring estimation of plant gradients, which makes this scheme very attractive for practical applications (Bunin et al., 2012). However, many process optimization problems do require gradient information to reach plant optimality. Therefore, in such cases, modifier adaptation (MA) that additionally corrects model gradients becomes more attractive RTO scheme. Nonetheless, upon convergence, CA guarantees finding a feasible input value for the plant constraints.

2.4 Influential Parameters

To understand the relationship between the model outputs (such as the cost and the constraint functions) and the model parameters, we introduce the concept of influential and non-influential parameters. Consider the mapping $y =$

$f(\boldsymbol{\theta})$, $y \in \mathbb{R}$, $\boldsymbol{\theta} \in \mathbb{R}^{n_\theta}$. In addition, consider the spaces $\boldsymbol{\mathcal{I}}$ and $\boldsymbol{\mathcal{N}\mathcal{I}}$ that are orthogonal complements to each other such that $\boldsymbol{\mathcal{I}} \oplus \boldsymbol{\mathcal{N}\mathcal{I}} = \mathbb{R}^{n_\theta}$, where $\oplus$ is the direct sum. Then, the influential and non-influential spaces can be defined as

Definition 1. (Influential parameters (Smith, 2014)). On the space $\boldsymbol{\mathcal{N}\mathcal{I}}$, a parameter vector $\boldsymbol{\theta}$ is said to be non-influential for a function $f \in \mathbb{R}$ if $|f(\boldsymbol{\theta}) - f(\boldsymbol{\theta}^\star)| < \epsilon$ for all $\boldsymbol{\theta}$ and $\boldsymbol{\theta}^\star \in \boldsymbol{\mathcal{N}\mathcal{I}}$, where ϵ is a small positive scalar. The orthogonal complement of $\boldsymbol{\mathcal{N}\mathcal{I}}$ is the space $\boldsymbol{\mathcal{I}}$ of influential parameters.

The influential and non-influential parameter spaces can be found via active subspaces (Constantine et al., 2014).

2.4.1 Active Subspaces. Active subspace methods determine the directions in parameter space that are most influential, that is, the directions that cause most variability in a model output (Constantine, 2015).

To find the active subspace for a given model output, the following matrix $\boldsymbol{C} \in \mathbb{R}^{n_\theta \times n_\theta}$ is evaluated:

$$\boldsymbol{C} = \int \big(\nabla_{\boldsymbol{\theta}} f(\boldsymbol{\theta})\big)^{\mathsf{T}} \big(\nabla_{\boldsymbol{\theta}} f(\boldsymbol{\theta})\big) \rho \, \mathrm{d}\boldsymbol{\theta}, \tag{6}$$

where $\nabla_{\boldsymbol{\theta}} f(\boldsymbol{\theta}) \in \mathbb{R}^{1 \times n_\theta}$ is the gradient vector of f with respect to $\boldsymbol{\theta}$, ρ is the probability density function of $\boldsymbol{\theta}$ over the admissible bounded set $\boldsymbol{\Theta} \subset \mathbb{R}^{n_\theta}$ with $\rho = 0$ for $\theta \notin \boldsymbol{\Theta}$. Here, it is assumed that $\boldsymbol{\theta}$ is the vector of normalized parameters that are scaled according to their range. Note that, since $\boldsymbol{C}$ is symmetric and positive semi-definite, it diagonalizes as:

$$\boldsymbol{C} = \boldsymbol{Q} \boldsymbol{\Pi} \boldsymbol{Q}^{\mathsf{T}}, \quad \boldsymbol{\Pi} = \mathrm{diag}(\pi_1, \ldots, \pi_{n_\theta}), \tag{7a}$$

with $\pi_1 \geq \cdots \geq \pi_{n_\theta} \geq 0$; $\boldsymbol{Q} \in \mathbb{R}^{n_\theta \times n_\theta}$ is an orthonormal matrix whose columns $\mathbf{q}_1, \ldots, \mathbf{q}_{n_\theta}$ are the normalized eigenvectors of $\boldsymbol{C}$.

It is shown in (Constantine, 2015) that, if the l^{th} eigenvalue π_l is zero, then the directional derivative of the scalar function f along the corresponding eigenvector is zero everywhere in the domain $\boldsymbol{\Theta}$. That is,

$$\big(\nabla_{\boldsymbol{\theta}} f(\boldsymbol{\theta})\big)\, \mathbf{q}_l = 0 \quad \forall\, \boldsymbol{\theta} \in \Theta. \tag{8}$$

Hence, when different components of the vector $\boldsymbol{\theta}$ are perturbed, the function f remains constant along the direction $\mathbf{q}_l$.

If there exists a sufficient gap in the eigenvalue spectrum, the following partitioning can be obtained,

$$\boldsymbol{\Pi} = \begin{bmatrix} \boldsymbol{\Pi}_1 & \mathbf{0} \\ \mathbf{0} & \boldsymbol{\Pi}_2 \end{bmatrix}, \text{ and } \boldsymbol{Q} = [\boldsymbol{Q}_1 \quad \boldsymbol{Q}_2], \tag{9}$$

where $\boldsymbol{\Pi}_1 \in \mathbb{R}^{m \times m}$ and $\boldsymbol{Q}_1 \in \mathbb{R}^{n_\theta \times m}$ with $m \leq n_\theta$. The block form is such that $m \ll n_\theta$ and $\pi_{m+1} \ll \pi_m$. Based on this partitioning, we have the new rotated coordinate variables $\boldsymbol{\vartheta}_1 \in \mathbb{R}^m$ and $\boldsymbol{\vartheta}_2 \in \mathbb{R}^{n_\theta - m}$ that are defined as:

$$\boldsymbol{\vartheta}_1 := (\boldsymbol{Q}_1)^{\mathsf{T}} \boldsymbol{\theta}, \text{ and } \boldsymbol{\vartheta}_2 := (\boldsymbol{Q}_2)^{\mathsf{T}} \boldsymbol{\theta}. \tag{10}$$

On average, the function f varies more dominantly due to parametric variability and perturbations in the directions described by $\boldsymbol{\vartheta}_1$ than in the directions described by $\boldsymbol{\vartheta}_2$. The influential and non-influential spaces are given by the column spaces of the matrices $\boldsymbol{Q}_1$ and $\boldsymbol{Q}_2$, respectively.

$$\boldsymbol{\mathcal{I}} = \mathrm{col}\,\big(\boldsymbol{Q}_1\big), \text{ and } \boldsymbol{\mathcal{N}\mathcal{I}} = \mathrm{col}\,\big(\boldsymbol{Q}_2\big), \tag{11}$$

where $\mathrm{col}(\cdot)$ is the column space.

Often, models are so complex that an analytical representation of the matrix $\boldsymbol{C}$ is not possible. In such cases, the influential and non-influential spaces are computed by approximating the matrix $\boldsymbol{C}$. The approximation $\hat{\boldsymbol{C}}$ is obtained by collecting N random samples of parameter $\boldsymbol{\theta}$ from the admissible set $\boldsymbol{\Theta}$ using the probability density ρ. For each sample i, the function gradient $\nabla_\theta f(\boldsymbol{\theta}_i)$ is approximated and constitutes the matrix $\hat{\boldsymbol{C}}$:

$$\hat{\boldsymbol{C}} = \frac{1}{N} \sum_{j=1}^{N} \big(\nabla_{\boldsymbol{\theta}} f(\boldsymbol{\theta}_j)\big)^{\mathsf{T}} \big(\nabla_{\boldsymbol{\theta}} f(\boldsymbol{\theta}_j)\big). \tag{12}$$

The gradient values can be obtained by finite-differences or by some of the Morris screening techniques presented in (Lewis et al., 2016). The algorithm for computing active subspaces via random sampling is detailed in (Constantine, 2015).

3 MODEL ADEQUACY FOR CONSTRAINT TRACKING

Certain processes are such that their optimal inputs $\mathbf{u}_p^\star$ are determined solely by active constraints. For such processes, the goal of the model-based optimization problem (2) is to match the plant constraints by iteratively adapting the model parameters. If there exist parameter values such that the model constraints match the plant constraints in the whole input space, then it is possible to compute the plant optimum via parameter adaptation and optimization. Such a model is said to be globally structurally correct. If not impossible, it is challenging to verify the global structural correctness of the model. This would require enormous experimental data with sufficient information to verify that there exist values of the model parameter $\boldsymbol{\theta}$ such that the model constraints match the plant constraints in the whole input space.

On the other hand, the modifier vector in (5) is tailored to achieve plant-model matching of the constraints. The addition of modifiers makes the model-based optimization problem (4) locally structurally correct. Expressed differently, at the current operating point $\mathbf{u}_k$, there exists modifier values such that the constraint values for the model and the plant match. Similarly, if there exists a set of model parameters that make the model locally structurally correct, then such a set can be adapted instead of the modifiers. Hence, we search for a set of model parameters, whose effect on the constraints is similar to that of the modifiers.

This would require the model parameters to mirror the behavior of the modifier $\boldsymbol{\varepsilon}_k^{\mathrm{G}}$. The behavior of the modifier $\boldsymbol{\varepsilon}_k^{\mathrm{G}}$ is captured by the following sensitivity equations:

$$\frac{\partial \Phi_{m,k}}{\partial \boldsymbol{\varepsilon}_k^{\mathrm{G}}} = \mathbf{0}_{1 \times n_{\mathrm{g}}}, \tag{13a}$$

$$\frac{\partial \mathbf{G}_{m,k}}{\partial \boldsymbol{\varepsilon}_k^{\mathrm{G}}} = \boldsymbol{I}_{n_{\mathrm{g}}}, \tag{13b}$$

where $\boldsymbol{I}$ is the identity matrix. Hence, it is desirable to search for a set of mirror parameters, $\boldsymbol{\theta}^{\mathrm{G}} \in \mathbb{R}^{n_{\mathrm{g}}}$, such that, at the given operating point $\mathbf{u}_k$, we have

$$\frac{\partial \Phi}{\partial \boldsymbol{\theta}^{\mathrm{G}}} \approx \mathbf{0}_{1 \times n_{\mathrm{g}}}, \tag{14a}$$

$$\frac{\partial \mathbf{G}}{\partial \boldsymbol{\theta}^{\mathrm{G}}} \approx \mathrm{diag}\,\big(\bar{G}_{1,k}(\theta_1^{\mathrm{G}}), \ldots, \bar{G}_{n_{\mathrm{g}},k}(\theta_{n_{\mathrm{g}}}^{\mathrm{G}})\big), \tag{14b}$$

where diag $(\cdot)$ represents a diagonal matrix with the diagonal entries $\bar{G}_{i,k} : \mathbb{R} \to \mathbb{R}$ and θ_i^{G} is the i^{th} element of the vector $\boldsymbol{\theta}^{\mathbf{G}}$. The advantage of the mirror parameters $\boldsymbol{\theta}^{\mathbf{G}}$ is that each of the constraint function G_i can be manipulated independently by the corresponding component θ_i^{G}.

To determine the vector of mirror parameters $\boldsymbol{\theta}^{\mathbf{G}}$, the parametric dependencies of the cost and constraint functions are investigated via influential spaces. At the operating point $\mathbf{u}_k$, the set $\boldsymbol{\mathcal{D}}_k^{(\mathrm{G}_i)}$ is constructed for the constraint G_i. This set is formed by finding the parameter directions that are simultaneously present in the influential space of G_i and in the non-influential spaces of the cost and the rest of the constraints.

$$\boldsymbol{\mathcal{D}}_k^{(\mathrm{G}_i)} := \boldsymbol{\mathcal{I}}_k^{(\mathrm{G}_i)} \bigcap \boldsymbol{\mathcal{NI}}_k^{(\Phi)} \bigcap_{j=1}^{j=n_{\mathrm{g}}} \boldsymbol{\mathcal{NI}}_k^{(\mathrm{G}_j)}, \quad j \neq i, \tag{15}$$

where $\boldsymbol{\mathcal{NI}}_k^{(\Phi)}$ is the non-influential parameter space of the cost function Φ; $\boldsymbol{\mathcal{I}}_k^{(\mathrm{G}_i)}$ and $\boldsymbol{\mathcal{NI}}_k^{(\mathrm{G}_i)}$ are the influential and non-influential parameter spaces of the constraint function G_i, respectively. Any direction in the set $\boldsymbol{\mathcal{D}}_k^{(\mathrm{G}_i)}$ is such that it affects only the corresponding constraint and not the cost and the rest of the constraints. Hence, this direction can be used to construct the vector $\boldsymbol{\theta}^{\mathbf{G}}$ that (approximately) satisfies (14). If the set $\boldsymbol{\mathcal{D}}_k^{(\mathrm{G}_i)}$ is non-empty, then θ_i^{G} can be constructed as

$$\theta_i^{\mathrm{G}} = (\mathbf{d}_k^{(\mathrm{G}_i)})^{\mathsf{T}} \boldsymbol{\theta}, \quad \mathbf{d}_k^{(\mathrm{G}_i)} \in \boldsymbol{\mathcal{D}}_k^{(\mathrm{G}_i)}. \tag{16}$$

Remark 1. The inactive (non-influential) subspaces are not necessarily formed by the eigenvectors with zero eigenvalues. Hence, parametric perturbations along the direction $\mathbf{d}_k^{(\mathrm{G}_i)}$ may still influence the cost and constraint functions other than G_i. However, since the eigenvalues corresponding to the inactive subspaces are relatively small, this influence is negligible. Moreover, some directions may not strictly belong to the set $\boldsymbol{\mathcal{D}}_k^{(\mathrm{G}_i)}$. These directions can still be used to form mirror parameters if the Euclidean norm of their projections on each of the influential and non-influential spaces in (15) is relatively larger than their projection on the respective complementary spaces.

In the case where the set $\boldsymbol{\mathcal{D}}_k^{(\mathrm{G}_i)}$ is empty, the model is unable to independently track the corresponding constraint via parameter adaptation. A zeroth-order modifier term can be added to that constraint to enable independent tracking.

4 PROPOSED ADAPTATION STRATEGY

For the purpose of tracking constraints, we propose a two-layer iterative approach. The top layer consists of model-based optimization with the inputs $\mathbf{u}$ as decision variables, while the bottom layer is the parameter estimation layer.

At the input value obtained in the optimization layer, the direction sets in (15) are computed. One or more constraints may have empty direction sets. Then, a constraint is selected that requires the largest number of non-influential spaces to be dropped from (15) before its direction set becomes non-empty. A zeroth-order modifier is used to track the selected constraint instead of mirror parameter. Subsequently, its non-influential space is reset to $\mathbb{R}^{n_\theta}$ and the direction sets $\boldsymbol{\mathcal{D}}_k^{(\mathrm{G}_i)}$ for the rest of the constraints are re-computed. The procedure is repeated until each constraint either has a modifier or has a non-empty set $\boldsymbol{\mathcal{D}}_k^{(\mathrm{G}_i)}$ for tracking. The estimation step minimizes the Euclidean norm of those constraints for which the set $\boldsymbol{\mathcal{D}}_k^{(\mathrm{G}_i)}$ is not empty. The mirror parameters resulting from such directions are the decision variables for the estimation step. The modifier values for the constraints with empty direction sets are computed at the updated parameter value obtained from parameter estimation. The updated parameter values and the modifiers are sent to the top layer. The two-layer procedure is repeated until convergence. It is illustrated next via a simple numerical example

$$\Phi(\mathbf{u}, \boldsymbol{\theta}) = \theta_3(u_1 + u_2) \tag{17a}$$
$$\mathrm{G}_1(\mathbf{u}, \boldsymbol{\theta}) = (\theta_1 + 3\,\theta_2)\, u_2 \leq 0 \tag{17b}$$
$$\mathrm{G}_2(\mathbf{u}, \boldsymbol{\theta}) = (2\,\theta_1 + 4\,\theta_2)\, u_1 + \theta_3\, u_2 \leq 0, \tag{17c}$$

with $\mathbf{u} = [u_1,\, u_2]^{\mathsf{T}}$ and $\boldsymbol{\theta} = [\theta_1,\, \theta_2,\, \theta_3]^{\mathsf{T}}$. At $\mathbf{u}_1 = [1,\, 1]^{\mathsf{T}}$, the constraints G_1 and G_2 exhibit the influential parameter spaces $[1,\, 3,\, 0]^{\mathsf{T}}$ and $[2,\, 4,\, 1]^{\mathsf{T}}$, respectively. Since the influential space of G_2 interacts with the influential spaces of the cost Φ and the constraint G_1, the zeroth-order modifier $\varepsilon_{2,k}^{\mathrm{G}}$ is added to the constraint G_2 and we set its non-influential space to $\mathbb{R}^3$. The influential space of the constraint G_1 does not interact with the influential space of the cost. Therefore, constraint G_1 is enforced via parameter adaptation within its influential space. Although adapting parameters along the parameter direction $[1,\, 3,\, 0]^{\mathsf{T}}$ also influences the constraint G_2, this influence is compensated by the modifier $\varepsilon_{2,k}^{\mathrm{G}}$ that independently tracks the constraint. The parameter estimation step at the 1^{st} RTO iteration for the numerical example then reads

$$(\theta_{1,k}^{\mathrm{G}}) := \underset{\theta_1^{\mathrm{G}}}{\operatorname{argmin}} \; ||\mathrm{G}_{p,1}(\mathbf{u}_1) - \mathrm{G}_1(\mathbf{u}_1, \boldsymbol{\theta})||^2,$$
$$\text{with } \theta_1^{\mathrm{G}} := [1,\, 3,\, 0]^{\mathsf{T}} \boldsymbol{\theta}.$$

In the estimation step, the parameters directions that are orthogonal to $[1,\, 3,\, 0]^{\mathsf{T}}$ are fixed at their current value. The adapted parameter $\boldsymbol{\theta}_1$ is obtained by transforming $\theta_{1,k}^{\mathrm{G}}$ and the fixed directions back to the original coordinates. The updated parameter $\boldsymbol{\theta}_1$ is then utilized in the optimization step with the modifier $\varepsilon_{2,2}^{\mathrm{G}}$ to obtain $\mathbf{u}_2$. At $\mathbf{u}_2$, the influential parameter spaces of the cost and constraints are re-computed via active subspaces, and the procedure is repeated until convergence. Note that it is recommended to control (limit) the step length for both the optimization and the parameter estimation steps to avoid oscillatory behavior.

5 SIMULATION STUDY

We investigate the run-to-run optimization of the semi-batch reactor originally proposed by Ruppen et al. (1997). The reactions occuring in the diketene-pyrrole reaction system are

$$A + B \xrightarrow{\mathsf{k}_1} C, \quad 2B \xrightarrow{\mathsf{k}_2} D, \quad B \xrightarrow{\mathsf{k}_3} E, \quad C + B \xrightarrow{\mathsf{k}_4} F.$$

Structural plant-model mismatch is considered by ignoring the third and fourth reactions, that is, by taking $\mathsf{k}_3 = 0$ and $\mathsf{k}_4 = 0$ in the model used for optimization. In addition, it is assumed that the model parameters k_1 and

Table 1. Kinetic parameter values for the diketene-pyrrole reaction

Parameter	Plant value	Nominal value	Model uncertainty	Probability distribution
k_1($\mathrm{L\,mol^{-1}min^{-1}}$)	0.053	0.053	[0.0011, 0.2120]	uniform
k_2($\mathrm{L\,mol^{-1}min^{-1}}$)	0.128	0.128	[0.0026, 0.5120]	uniform
k_3($\mathrm{min^{-1}}$)	0.028	0	-	-
k_4($\mathrm{L\,mol^{-1}min^{-1}}$)	0.001	0	-	-

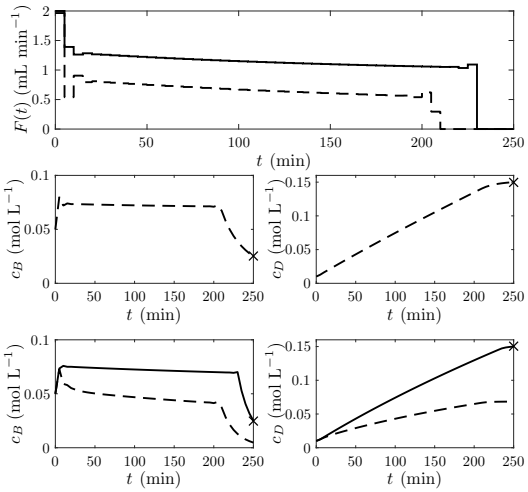

Fig. 1. Diketene-pyrrole reaction. **Top plots:** Optimal input profile (solid line: plant optimum; dashed line: model optimum). **Middle plots:** Model constraints with the model optimal input. **Bottom plots:** Solid lines: plant constraints with the plant optimal input. Dashed lines: plant constraints with the model optimal input.

k_2 are uncertain, and follow a uniform distribution. The parameter values used to represent both the plant and the model are listed in Table 1. The objective is to maximize the yield of product C, while meeting input constraints as well as terminal constraints on the concentrations of the reactant B and the side product D:

$$\max_{F(t)} \quad J := c_C(t_f)V(t_f) \tag{18}$$
$$\text{s.t.} \quad \text{model in (Chachuat et al., 2009)},$$
$$c_B(t_f) \le c_B^{max},\ c_D(t_f) \le c_D^{max},\ 0 \le F(t) \le F^{max},$$

where c_A, c_B, c_C and c_D represent the concentrations of the species A, B, C and D, respectively. V is the reactor volume, F is the inlet flowrate of species B and c_B^{in} is the concentration of B in the feed. The values of the bounds c_B^{max} and c_D^{max} are 0.025 and 0.15, respectively. The values of the other parameters and of the initial concentrations are taken from Chachuat et al. (2009).

The NLP formulated to solve this optimization problem contains 50 time stages. Each stage corresponds to piecewise-constant values of the input $F(t)$. The profiles of the optimal input for both the plant and the model (at the nominal parameter value) are shown in the top plot of the Figure 1. Clearly, the optimal profiles obtained for the model and the plant are very different. Model constraints at the model optimal input are shown in the middle plots of Figure 1. The bottom plots of Figure 1 shows the plant constraints for both the plant and the model optimal inputs. As can be seen in Figure 1, the concentration of reactant B at final time is not at its bound when the model optimal input is applied, whereas it lies at the bound when the plant optimal input is applied. The plant yield of product C obtained with the model optimal input is 0.3875, which is much lower than the plant optimal yield of 0.5081.

The input and output profiles reached upon convergence for the two-step approach, constraint adaptation and the proposed approach are shown in Figure 2. The two-step approach leads to an improvement in the yield of C from one batch to another but converges to a considerably sub-optimal value of 0.4414, whereas constraint adaptation leads to a near optimal yield of 0.5076. The small optimality gap is the result of not adapting the gradient modifiers. The proposed adaptation strategy converges to the plant optimum.

In the proposed strategy, a single parameter, which is a linear combination of k_1 and k_2, is adapted to track the terminal constraint on c_B. This linear combination is such that (i) it has a large norm when projected on the influential space of the terminal constraint on c_B and thus can be used for tracking this constraint, and (ii) it lies perfectly in the non-influential space of the cost of Problem (18) and thus will not affect the cost. However, this direction does affect the terminal constraint on c_D, which implies that, when the linear combination of parameters is adapted, it also influences the terminal constraint on c_D. However, since the terminal constraint on c_D is tracked by a modifier, the influence of parameter adaptation on this constraint is irrelevant. Note that no modifier is added to the constraint on c_B. The evolution of the model parameters over the RTO iterations is shown in Figure 3.

6 CONCLUSIONS

This paper has highlighted various model correction strategies to determine optimal operating conditions by exploiting process data. The advantages and disadvantages of model parameter adaptation compared to modifier adaptation have been discussed. To capitalize on the advantages of both, a novel adaptation strategy has been proposed that searches for parameters capable of mirroring the effect of modifiers. Modifiers are then added only when such mirror parameters are not found. A simulation study of run-to-run batch optimization illustrates that a combination of mirror parameters and modifiers can successfully track optimality conditions.

References

Ahmad, A., Gao, W., and Engell, S. (2017). Effective model adaptation in iterative rto. In *27th European Symposium on Computer Aided Process Engineering*, volume 40 of *Computer Aided Chemical Engineering*, 1717 – 1722. Elsevier.

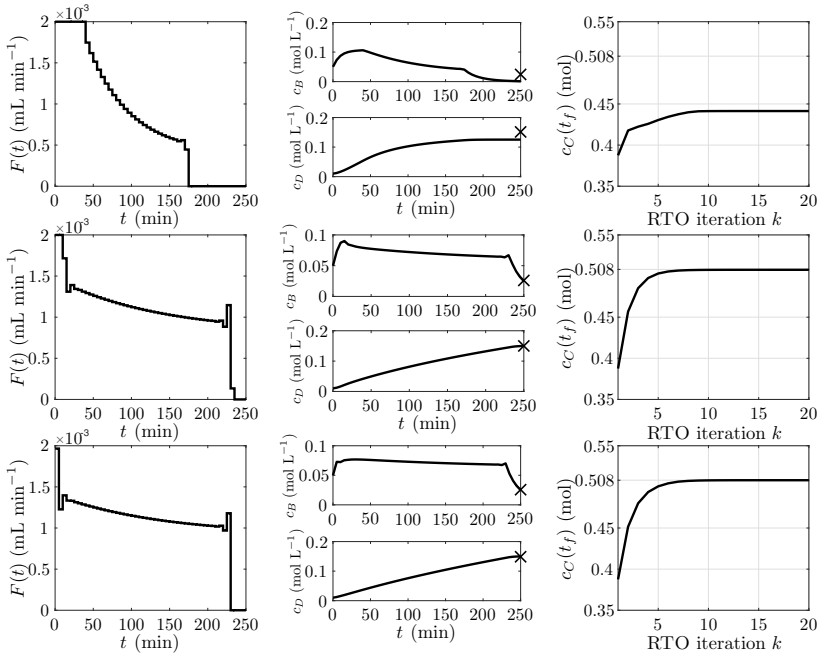

Fig. 2. Application of different adaptation strategies to Problem (18). **Top plots:** Two-step approach **Middle plots:** Constraint adaptation **Bottom plots:** Proposed approach.

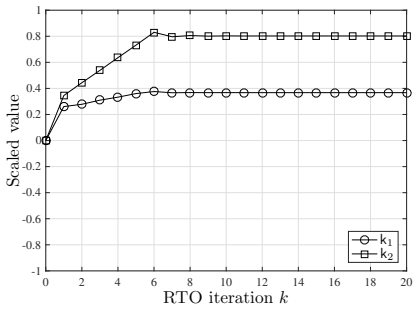

Fig. 3. Model parameters evolution over RTO iterations.

Brdyś, M. and Tatjewski, P. (2005). *Iterative Algorithms for Multilayer Optimizing Control*. Imperial College Press, London, UK.

Bunin, G.A., Wuillemin, Z., François, G., Nakajo, A., Tsikonis, L., and Bonvin, D. (2012). Experimental real-time optimization of a solid oxide fuel cell stack via constraint adaptation. *Energy*, 39(1), 54 – 62.

Chachuat, B., Marchetti, A., and Bonvin, D. (2008). Process optimization via constraints adaptation. *J. Process Contr.*, 18(3), 244–257.

Chachuat, B., Srinivasan, B., and Bonvin, D. (2009). Adaptation strategies for real-time optimization. *Comp. Chem. Engng.*, 33(10), 1557–1567.

Chen, C.Y. and Joseph, B. (1987). On-line optimization using a two-phase approach: An application study. *Ind. Eng. Chem. Res.*, 26(9), 1924–1930.

Constantine, P.G. (2015). *Active Subspaces: Emerging Ideas for Dimension Reduction in Parameter Studies*. SIAM, Philadelphia, PA, USA.

Constantine, P.G., Dow, E., and Wang, Q. (2014). Active subspace methods in theory and practice: Applications to kriging surfaces. *SIAM Journal on Scientific Computing*, 36(4), A1500–A1524.

Costello, S., François, G., and Bonvin, D. (2016). A directional modifier-adaptation algorithm for real-time optimization. *J. Process Contr.*, 39, 64 – 76.

Forbes, J., Marlin, T., and MacGregor, J. (1994). Model adequacy requirements for optimizing plant operations. *Comp. Chem. Engng.*, 18(6), 497–510.

Gao, W. and Engell, S. (2005). Iterative set-point optimization of batch chromatography. *Comp. Chem. Engng.*, 29(6), 1401–1409.

Gao, W., Wenzel, S., and Engell, S. (2016). A reliable modifier-adaptation strategy for real-time optimization. *Comp. Chem. Engng.*, 91, 318–328.

Lewis, A., Smith, R.C., and Williams, B. (2016). Gradient free active subspace construction using morris screening elementary effects. *Comp. Math. App.*, 72(6), 1603 – 1615.

Mandur, J. and Budman, H. (2015). Simultaneous model identification and optimization in presence of model-plant mismatch. *Chem. Engng. Sci.*, 129(10), 106–115.

Marchetti, A. (2009). *Modifier-Adaptation Methodology for Real-Time Optimization*. EPFL thesis, # 4449, Ecole Polytechnique Fédérale de Lausanne. doi:10.5075/EPFL thesis-4449.

Marchetti, A., Chachuat, B., and Bonvin, D. (2009). Modifier-adaptation methodology for real-time optimization. *Ind. Eng. Chem. Res.*, 48(13), 6022–6033.

Marchetti, A., François, G., Faulwasser, T., and Bonvin, D. (2016). Modifier adaptation for real-time optimization – Methods and applications. *Processes*, 4(55), 1–35.

Ruppen, D., Bonvin, D., and Rippin, D.W.T. (1997). Implementation of adaptive optimal operation for a semi-batch reaction system. *Comp. Chem. Engng.*, 22(1-2), 185–199.

Singhal, M., Marchetti, A., Faulwasser, T., and Bonvin, D. (2018). Active directional modifier adaptation for real-time optimization. *Comp. Chem. Engng. (In press)*.

Singhal, M., Marchetti, A., Faulwasser, T., and Bonvin, D. (2017). Improved directional derivatives for modifier-adaptation schemes. *IFAC-PapersOnLine*, 50(1), 5718 – 5723. 20th IFAC World Congress.

Smith, R.C. (2014). *Uncertainty Quantification: Theory, Implementation, and Applications*. SIAM Computational Science & Engineering Series: Philadelphia, USA.

Yip, W. and Marlin, T. (2004). The effect of model fidelity on real-time optimization performance. *Comp. Chem. Engng.*, 28, 267–280.